

Read-write method for flash memory chip

Publication number: CN1362708
Publication date: 2002-08-07
Inventor: YUE JINGXING (CN); WU XIULIN (CN)
Applicant: WU XIULI (CN)
Classification:
- International: G11C7/00; G11C7/00; (IPC1-7): G11C7/00
- European:
Application number: CN20011000002 20010102
Priority number(s): CN20011000002 20010102

Report a data error here

Abstract of CN1362708

The present invention relates to a flash chip reading and writing method. Every storage block in the interior of flash chip respectively possesses data area and control area, the control area can be divided into index zone, logical address pointer zone and state identification zone. According to the information described by said three zone the data area of flashed every storage block can be erased, read, written and undergone the process of data updating operation. Said invention can raise its integrated reading and writing speed, can make the frequencies of writing-in of every storage block in flash chip be identical basically to prolong the life of flash plate.

Data supplied from the **esp@cenet** database - Worldwide

Read-write method for flash memory chip

Publication number: CN1362708

Publication date: 2002-08-07

Inventor: YUE JINGXING (CN); WU XIULIN (CN)

Applicant: WU XIULI (CN)

Classification:

- international: **G11C7/00; G11C7/00; (IPC1-7): G11C7/00**

- European:

Application number: CN20011000002 20010102

Priority number(s): CN20011000002 20010102

Abstract of CN1362708

The present invention relates to a flash chip reading and writing method. Every storage block in the interior of flash chip respectively possesses data area and control area, the control area can be divided into index zone, logical address pointer zone and state identification zone.

According to the information described by said three zone the data area of flashed every storage block can be erased, read, written and undergone the process of data updating operation. Said invention can raise its integrated reading and writing speed, can make the frequencies of writing-in of every storage block in flash chip be identical basically to prolong the life of flash plate.

Data supplied from the esp@cenet database - Worldwide

G11C 7/00

[21] 申请号 01100002.3

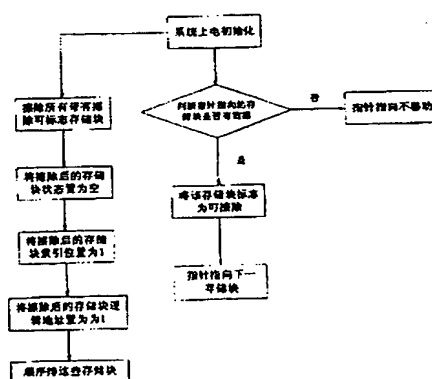
[11]公开号 CN 1362708A

[74] 专利代理机构 北京三友知识产权代理有限公司
代理人 刘芳

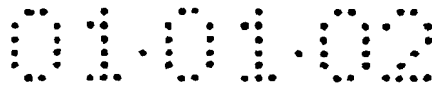
权利要求书 4 页 说明书 9 页 附图页数 6 页

[57] 摘要

一种闪存芯片的读写方法,闪存芯片内部的各存储块分有数据区 和控制区,控制区划分为索引区、逻辑地址指针区和状态标识区,根据索引区、逻辑地址指针区和状态标识区所表述的信息,对闪存 的各存储块的数据区进行擦除、读、写和数据更新操作。本发明根据存储块的状态进行擦除、读、写操作,将闪存的写入操作与擦除 操作分开,提高整体的读写速度,使得对闪存中各个存储块的写入 操作的频率基本一致,延长闪存盘的寿命。



ISSN 1008-4274



权 利 要 求 书

1、一种闪存芯片的读写方法，其特征在于：闪存芯片内部的各存储块分有数据区和控制区，控制区划分为索引区、逻辑地址指针区和状态标识区，根据索引区、逻辑地址指针区和状态标识区所表述的信息，对闪存的各存储块的数据区进行擦除、读、写和数据更新操作。

2、如权利要求 1 所述的闪存芯片的读写方法，其特征在于：所述的状态标识区标识所对应存储块工作状态的状态标识位组可为相对应的存储块 (Block) 的四种状态之一：“空”或“忙”或“释放”或“丢弃”。

3、如权利要求 1 所述的闪存芯片的读写方法，其特征在于：所述的索引区中的索引位组的信息为与之相对应的存储块使用状态之一：“未用过”或“已使用”。

4、如权利要求 1 所述的闪存芯片的读写方法，其特征在于：所述的逻辑地址指针区内的逻辑地址指针组信息为与之相对应的存储块的物理地址或表示多次更新后、按照更新次数顺序记录的逻辑地址。

5、如权利要求 1 或 2 或 3 或 4 所述的闪存芯片的读写方法，其特征在于：所述的擦除操作步骤为：

a、依次检测闪存盘中所有或设定数目的部分存储块状态标识的状态数据；

b、如果该状态为“释放”，则对与存储块状态标识相对应的存储块全块进行擦除；否则，进入步骤 e；

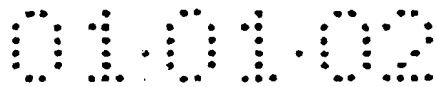
c、将该存储块的索引区中所有的索引信息设置为：“未用”；

d、将该存储块的逻辑地址指针区中所有的逻辑地址指针擦除；

e、如果该状态为“空闲”或“忙”或“丢弃”；

f、如果未检测完闪存盘中所有的或设定数据的部分存储块状态标识的状态数据，返回步骤 a；否则，进入步骤 g；

g、结束。



6、如权利要求 1 所述的闪存芯片的读写方法，其特征在于：所述的擦除操作可为初始化擦除和其他时间的擦除。

7、如权利要求 6 所述的闪存芯片的读写方法，其特征在于：所述的初始化擦除操作为在该闪存盘加电时所作的第一个操作。

5 8、如权利要求 6 所述的闪存芯片的读写方法，其特征在于：所述的其他时间擦除可为读数据块内文件的同时擦除状态为“释放”数据块。

9、如权利要求 1 所述的闪存芯片的读写方法，其特征在于：所述的擦除也可读操作擦除或操作写擦除。

10 10、如权利要求 9 所述的闪存芯片的读写方法，其特征在于：所述的读操作擦除为在每个读操作之前的擦除操作。

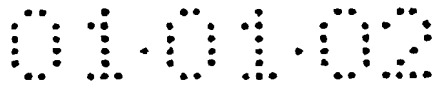
11、如权利要求 5 或 10 所述的闪存芯片的读写方法，其特征在于：所述的读操作擦除为：如果存储块状态为“释放”的存储块的数量少于一事先设置的最小的读操作擦除存储块的数量，将其与该事先设置的最小的读操作擦除存储块的数量之间的差值数量的“丢弃”的存储块设置为被擦除的对象，进行擦除操作。

12、如权利要求 9 所述的闪存芯片的读写方法，其特征在于：所述的写操作擦除为在每个写操作之前的擦除操作。

13、如权利要求 5 或 12 所述的闪存芯片的读写方法，其特征在于：所述的写操作擦除为：如果存储块状态为“释放”的存储块的数量少于一事先设置的最小的写操作擦除存储块的数量，将其与该事先设置的最小的读操作擦除存储块的数量之间的差值数量的“丢弃”的存储块设置为被擦除的对象，进行擦除操作。

14、如权利要求 1 或 2 或 3 或 4 所述的闪存芯片的读写方法，其特征在于：所述的读操作包括如下步骤：

25 a、根据给出的访问地址找到相应的存储块，依次检测该存储块中索引区各索引信息中的内容；



b、如果找到最后一个内容记录“已使用”的索引信息，进入步骤 c；
否则返回到步骤 a；

c、根据与该索引信息对应的逻辑地址指针中的地址找到该地址指示的存储块，读出该存储块内的数据内容，结束。

5 15、如权利要求 1 或 2 或 3 或 4 所述的闪存芯片的读写方法，其特征在于：所述的写操作包括如下步骤：

a、找到当前指针所指向的为“空闲”状态的存储块，将所要写的数据写入该存储块的数据区中；

10 b、再读回该存储块的数据内容，将步骤 a 所写的数据与读回的该存储块的数据内容进行比较；

c、如果比较结果为不一致，进入步骤 d；否则进入步骤 e；

d、将当前存储块的状态修改为“丢弃”，返回步骤 a；

e、将与当前存储块相对应的索引区中相应的状态内容设置为“已使用”；

15 f、将其相对应的逻辑地址指针的内容设置为该数据存入的存储块的地址；

g、将当前存储块的状态修改为“忙”，结束。

16、如权利要求 1 或 15 所述的闪存芯片的读写方法，其特征在于：所述的数据更新操作包括如下步骤：

20 a、根据给出的被更新数据的地址找到相应的存储块，分析与之相对应的状态标识区中的信息内容；

b、如果该状态标识区中的内容为“忙”，依次检测该存储块中索引区各索引信息中的内容，直到找到最后一个内容值记录为“已使用”的索引信息；

25 c、根据步骤 b 中找到的索引信息对应的逻辑地址指针的内容，找到该逻辑地址指针所指向的存储块；

d、将步骤 c 中找到的存储块所对应的状态标识区中的内容设置为“释

放”;

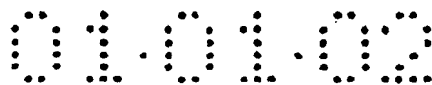
e、找到当前指针指向的首个为“空闲”状态的存储块,进行写操作。

f、将步骤 b 中最后一个内容值记录为“已使用”索引信息后面的一个索引信息的内容设置为“已使用”;将与该索引信息其相对应的逻辑地址指针的内容设置为步骤 e 所写数据的存储块的地址;

g、将步骤 a 中的被更新数据的地址找到相应的存储块相对应的状态标识区中的信息内容设置为“释放”;结束。

17、如权利要求 16 所述的闪存芯片的读写方法,其特征在于:所述的数据更新操作还包括如下步骤:将被更新数据的存储块内本次不更新的其他文件的内容进行复制,再将该文件内容搬移到当前指针更新写入的存储块相应位置上。

18、如权利要求 15 或 16 所述的闪存芯片的读写方法,其特征在于:所述的为“空闲”状态的存储块的地址存储在闪存盘的一连续的存储区中。



说明书

一种闪存芯片的读写方法

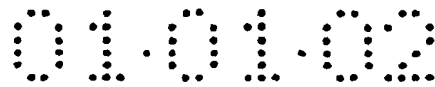
5 本发明涉及一种闪存芯片的读写方法，尤指一种针对以闪存芯片为基本存储介质的闪存盘的数据写入和读出的方法。

众所周知，现有的闪存均是由存储块组成的，每一个存储块内均包含两个区域：数据存储区 100 和存储块状态区 500；数据存储区用于实际的数据存储，存储块状态区内则是标识存储块内的存储块好坏的存储块的状态信息。如果，
10 存储块内的存储块均是正常的，存储块的状态信息被设定为“好”；一旦存储块内的存储块出现坏的单元，存储块的状态信息被设定为“坏”。通过上述的设定，闪存的读写管理软件就可以在对闪存进行读写操作时方便地“避开”被设定为“坏”的存储块。另外，一般对闪存进行数据更新操作，需要先擦除原有数据，然后才能写入新的数据。

15 上述现有的闪存的存储块的分区管理方式和数据的更新方式是一种基本的操作方式。但是上述方式存在着一些缺点：首先，存储块的状态区只有“好”和“坏”两种状态，不利于描述其所对应的存储块的更详细的工作状态和使用信息；其次，由于数据更新操作需要先擦除原有数据，才能写入新的数据，因此，数据更新操作所花费的时间远比对闪存进行写操作所花费的时间要多得多；再有，现有的数据更新操作不对所写入的存储块的使用频率进行统计，因此会造成闪存中各个存储块老化速度不均匀，降低了闪存的使用寿命。
20

本发明的主要目的在于提供一种闪存芯片的读写方法，将闪存盘的闪存中的存储块进行管理，增加状态标识，根据存储块的状态进行擦除、读、写操作，从而将闪存的写入操作与擦除操作分开，提高整体的读写速度，避免在向闪存
25 盘写入数据耗时过长，预防由于系统致命的瘫痪所造成的数据损失。

本发明的又一目的在于提供一种闪存芯片的读写方法，使得对闪存中各个



存储块的写入操作的频率基本一致，避免闪存中各个存储块的老化程度不均，实现所有的存储块都被均匀分配写入，最终延长闪存盘的寿命。

本发明的目的是通过如下技术方案实现的：

一种闪存芯片的读写方法，闪存芯片内部的各存储块分有数据区和控制区，控制区划分为索引区、逻辑地址指针区和状态标识区，根据索引区、逻辑地址指针区和状态标识区所表述的信息，对闪存的各存储块的数据区进行擦除、读、写和数据更新操作。

所述的状态标识区标识所对应存储块工作状态的状态标识位组可为相对应的存储块（Block）的四种状态之一：“空”或“忙”或“释放”或“丢弃”。

所述的索引区中的索引位组的信息为与之相对应的存储块使用状态之一：“未用过”或“已使用”。

所述的逻辑地址指针区内的逻辑地址指针组信息为与之相对应的存储块的物理地址或表示多次更新后、按照更新次数顺序记录的逻辑地址。

所述的擦除操作步骤为：

- 1、依次检测闪存盘中所有或设定数目的部分存储块状态标识的状态数据；
- 2、如果该状态为“释放”，则对与存储块状态标识相对应的存储块全块进行擦除；否则，进入步骤 5；
- 3、将该存储块的索引区中所有的索引信息设置为：“未用”；
- 4、将该存储块的逻辑地址指针区中所有的逻辑地址指针擦除；
- 5、如果该状态为“空闲”或“忙”或“丢弃”；
- 6、如果未检测完闪存盘中所有的或设定数据的部分存储块状态标识的状态数据，返回步骤 1；否则，进入步骤 7；
- 7、结束。

所述的擦除操作可为初始化擦除和其他时间的擦除。初始化擦除操为在该闪存盘加电时所作的第一个操作。其他时间擦除可为读数据块内文件的同时擦



除状态为“释放”数据块。擦除也可读操作擦除或操作写擦除，读操作擦除为在每个读操作之前的擦除操作；写操作擦除为在每个写操作之前的擦除操作。

5 所述的读操作擦除为：如果存储块状态为“释放”的存储块的数量少于一事先设置的最小的读操作擦除存储块的数量，将其与该事先设置的最小的读操作擦除存储块的数量之间的差值数量的“丢弃”的存储块设置为被擦除的对象，进行擦除操作。

10 所述的写操作擦除为：如果存储块状态为“释放”的存储块的数量少于一事先设置的最小的写操作擦除存储块的数量，将其与该事先设置的最小的读操作擦除存储块的数量之间的差值数量的“丢弃”的存储块设置为被擦除的对象，进行擦除操作。

所述的读操作包括如下步骤：

- 1、 根据给出的访问地址找到相应的存储块，依次检测该存储块中索引区各索引信息中的内容；
- 15 2、 如果找到最后一个内容记录“已使用”的索引信息，进入步骤 3；否则返回到步骤 1；
- 3、 根据与该索引信息对应的逻辑地址指针中的地址找到该地址指示的存储块，读出该存储块内的数据内容，结束。

所述的写操作包括如下步骤：

- 20 1、 找到当前指针所指向的为“空闲”状态的存储块，将所要写的数据写入该存储块的数据区中；
- 2、 再读回该存储块的数据内容，将步骤 1 所写的数据与读回的该存储块的数据内容进行比较；
- 3、 如果比较结果为不一致，进入步骤 4；否则进入步骤 5；
- 25 4、 将当前存储块的状态修改为“丢弃”，返回步骤 1；
- 5、 将与当前存储块相对应的索引区中相应的状态内容设置为“已使



用”;

- 6、 将与其相对应的逻辑地址指针的内容设置为该数据存入的存储块的地址;
- 7、 将当前存储块的状态修改为“忙”, 结束。

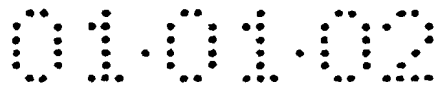
5 所述的数据更新操作包括如下步骤:

- 1、 根据给出的被更新数据的地址找到相应的存储块, 分析与之相对应的状态标识区中的信息内容;
- 2、 如果该状态标识区中的内容为“忙”, 依次检测该存储块中索引区各索引信息中的内容, 直到找到最后一个内容值记录为“已使用”的索引信息;
- 10 3、 根据步骤 2 中找到的索引信息对应的逻辑地址指针的内容, 找到该逻辑地址指针所指向的存储块;
- 4、 将步骤 3 中找到的存储块所对应的状态标识区中的内容设置为“释放”;
- 15 5、 找到当前指针指向的首个为“空闲”状态的存储块, 进行写操作。
- 6、 将步骤 2 中最后一个内容值记录为“已使用”索引信息后面的一个索引信息的内容设置为“已使用”; 将与该索引信息其相对应的逻辑地址指针的内容设置为步骤 5 所写数据的存储块的地址;
- 7、 将步骤 1 中的被更新数据的地址找到相应的存储块相对应的状态标识区中的信息内容设置为“释放”; 结束。
- 20

所述的数据更新操作还包括如下步骤: 将被更新数据的存储块内本次不更新的其他文件的内容进行复制, 再将该文件内容搬移到当前指针更新写入的存储块相应位置上。

所述的为“空闲”状态的存储块的地址存储在闪存盘的一连续的存储区中。

- 25 本发明通过将构成闪存盘的闪存中的存储块内的状态区的重新划分为索引区、逻辑地址指针区和存储块状态标识区, 实现了利用各个分区来存储与之对



应的存储块数据区的操作状态信息的目的，将闪存的写入操作与擦除操作分开，避免了写入数据耗时过长的缺点，可以预防由于系统以外瘫痪所造成的数据损失。

此外，本发明通过存储块内的状态区的重新划分，使每个闪存单元只能在清除状态时被写入，被写入的位将自然地保持到该位因为需要重写而将包括该位的整个存储块被擦除之前；最后，每个存储块的写操作均因为可成功地找到可写入的“空”存储块，而不需要先擦除它；由于每一个要写的存储块均在不同的逻辑存储块上，空闲状态存储块按照指针指向顺序排列，写入时是一一按次序被使用，因此没有哪个存储块可被连续地写入，使得对闪存中各个存储块的写入操作的频率基本一致，避免了闪存中各个存储块的老化程度不均，实现了所有的存储块都被均匀分配写入，最终延长了闪存盘的寿命。

以下结合附图和具体实施方案对本发明做进一步的详细说明：

图 1 为现有闪存中存储块的数据存储部分和坏块标识部分划分示意图。

图 2 为本发明对闪存中存储块的划分及状态信息标识的方法示意图。

15 图 3 为本发明对闪存中存储块首次写入数据后的状态示意图。

图 4 为本发明对闪存中存储块数据更新后的状态示意图。

图 5 为本发明对闪存中存储块写操作基本流程图；

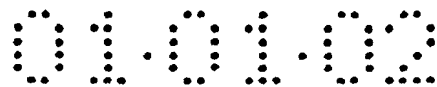
图 6 为本发明对闪存中存储块读操作基本流程图；

图 7 为本发明对闪存中存储块初始化及擦除过程流程图；

20 图 8 为本发明对闪存中存储块数据更新操作流程图中。

如图 1-4 所示，本发明闪存芯片内部的各存储块分有数据区和控制区，控制区划分为索引区、逻辑地址指针区和状态标识区，根据索引区、逻辑地址指针区和状态标识区所表述的信息，对闪存的各存储块的数据区进行擦除、读、写和数据更新操作。闪存中存储块管理方法为本发明方法的基础依据。

25 状态标识区标识所对应存储块工作状态的状态标识位组可为相对应的存储块（Block）的四种状态之一：“空”或“忙”或“释放”或“丢弃”。状态标



识区的状态标识位组数与数据块的数量一致。1 号数据块为坏块，该数据块的号码信息被另放入控制区不良存储信息内。

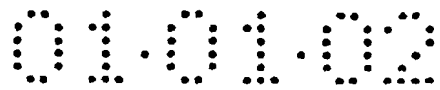
索引区中的索引位组的信息为与之相对应的存储块使用状态之一：“未用过或已使用”。其中“未用过”用“1”表示；“已使用”用“0”表示。

5 逻辑地址指针区内的逻辑地址指针组信息为与之相对应的存储块的物理地址号码。索引区中的索引位组的数量与逻辑地址指针区中的逻辑地址指针组的数量相等。

10 当数据块内没有写入数据为空闲状态时，索引位组的信息为“1”，状态标识位组可为“空”，表示该数据块为空闲状态，逻辑地址指针组信息没有地址指针设置，这是对应的状态标识位组为“空”，表示该数据块为空闲状态，可写入数据；当然，该数据块号码如果被表示在不良信息管理区内，表示该数据块为“坏的”，则不能写入数据。这一步往往是向数据块内写入数据时首先应该进行。

15 参见图 3，在该实施例中，2 号数据块写入了数据，这是其对应的索引位信息置为 0，逻辑地址指针的地址置为该数据块的物理地址 2，2 号数据块的状态被标识为“忙”。

20 参见图 4，当对应的 2 号数据块有更新操作，并且更新的数据放入数据块状态标识位组的标记为“空”（如图 3 所示）的 3 号数据块中后，2 号数据块索引位组增加一个已使用的标记“0”，对应的增加标识逻辑地址指针区内的逻辑地址指针组信息也增加一个更新次数顺序记录的最后一个数据块逻辑地址，故应为“3”。这时，相应的 2 号数据块状态标识位组的标记为“丢弃”；3 号数据块状态标识位组的标记为“忙”。在某时刻，由系统对标识为“丢弃”的块进行擦除，擦除完毕后，将该数据块状态标识位组的标记为“释放”。因此数据更新时只要将数据直接写入一个空的数据块中即可，而不需要先进行擦除的操作，节省了大量的时间，提高了闪存的读写速度。标识索引状态的索引
25 区中的索引位组的数量与逻辑地址指针区中的逻辑地址指针组的数量相等。其



索引位组的数量与逻辑地址指针区中的逻辑地址指针组的数量应为该数据块更新操作的次数。而状态标识区的状态标识位组的数量可与数据块的数量相等。

另外当遇到坏块时，可通过在对应的标识位组标记“坏”，将该坏块区分出来，置为不用，而不至于影响整个闪存片的使用。

5 如图 5 所示，本发明的写操作写操作包括如下步骤：

- 1、 找到当前指针所指向的为“空闲”状态的存储块 N，将所要写的数据写入存储块 N 的数据区中；
- 2、 再读回该存储块 N 的数据内容，将步骤 1 所写的数据与读回的该存储块 N 的数据内容进行比较；
- 10 3、 如果比较结果为不一致，将当前存储块 N 的状态修改为“丢弃”，将该存储块 N 的号码存入“坏块”的号码管理区域内，返回步骤 1；
- 4、 如果比较结果为一一致，则写入数据，将与当前存储块相对应的索引区中相应的状态内容设置为“已使用”；
- 5、 将与其相对应的逻辑地址指针的内容设置为该数据存入的存储块的地址；
- 15 6、 将当前存储块的状态修改为“忙”，结束。

参见图 6，本发明读操作包括如下步骤：

- 1、 根据给出的访问地址找到相应的存储块 M，依次检测该存储块 M 中索引区各索引信息中的内容；
- 20 2、 如果找到最后一个内容记录“已使用”即为“0”的索引信息，进入步骤 3；否则返回到步骤 1；
- 3、 根据与该索引信息对应的逻辑地址指针中的地址找到该地址指示的存储块 N，读出该存储块 N 内的数据内容 A，
- 4、 完成整个读的操作。

25 图 7 所示的是本发明擦除操作的方法。本发明的一个重要目的就是将擦除操作与写操作分开，以提高整个读写的速度。因此本发明的擦除操作可为初始



化擦除和其他时间的擦除。而初始化擦除操为在该闪存盘加电时所作的第一个操作；其他时间擦除可为读数据块内文件的同时擦除状态为“释放”数据块，这样将改变现有技术中必须再完成了擦除操作后，才能进行写操作的缺点。

具体擦除操作步骤为：

- 5 1、 依次检测闪存盘中所有或设定数目的部分存储块状态标识的状态数据；
- 2、 如果该状态为“释放”，则对与存储块状态标识相对应的存储块全块进行擦除；否则，进入步骤 5；
- 3、 将该存储块的索引区中所有的索引信息设置为：“未用”；
- 10 4、 将该存储块的逻辑地址指针区中所有的逻辑地址指针擦除；
- 5、 如果该状态为“空闲”或“忙”或“丢弃”；
- 6、 如果未检测完闪存盘中所有的或设定数据的部分存储块状态标识的状态数据，返回步骤 1；否则，进入步骤 7；
- 7、 结束。

15 当然，擦除也可读操作擦除或操作写擦除。而读操作擦除为在每个读操作之前的擦除操作，具体是依据：如果存储块状态为“释放”的存储块的数量少于一事先设置的最小的读操作擦除存储块的数量，将其与该事先设置的最小的读操作擦除存储块的数量之间的差值数量的“丢弃”的存储块设置为被擦除的对象，进行擦除操作。

20 对于写操作擦除，也为在每个写操作之前的擦除操作。如果存储块状态为“释放”的存储块的数量少于一事先设置的最小的写操作擦除存储块的数量，将其与该事先设置的最小的读操作擦除存储块的数量之间的差值数量的“丢弃”的存储块设置为被擦除的对象，进行擦除操作。

25 另外，为避免系统意外断电，而将正在进行写操作，存储块中已经有了数据，但是该块的状态尚未改变，因此再上电时，首先应判断指针所指向的存储块内是否有数据。如果有数据，并检测该存储块的状态为“空闲”时，则该存



储块标记为”释放“，指针向下移动。否则指针指向不改变。

如图 8 所示为本发明数据更新的方法。

它包括如下步骤：

- 1、 根据给出的被更新数据的地址找到相应的存储块，分析与之相对应的
5 的状态标识区中的信息内容 A；
- 2、 如果该状态标识区中的内容为“忙”，依次检测该存储块中索引区
各索引信息中的内容，直到找到最后一个内容值记录为“已使用”
的索引信息；
- 3、 根据步骤 2 中找到的索引信息对应的逻辑地址指针的内容，找到该
10 逻辑地址指针所指向的存储块 M；
- 4、 将步骤 3 中找到的存储块所对应的状态标识区中的内容设置为“释
放”，即为可擦除；
- 5、 找到当前指针指向的首个为“空闲”状态的存储块 N，进行写操作。
- 6、 将步骤 2 中最后一个内容值记录为“已使用”索引信息后面的一个
15 索引信息的内容设置为“已使用”；将与该索引信息其相对应的逻
辑地址指针的内容设置为步骤 5 所写数据的存储块的地址 N；
- 7、 将步骤 1 中的被更新数据的地址找到相应的存储块 M 相对应的状态
标识区中的信息内容设置为“释放”；结束。

另外，当存储块中有一个以上不同的文件时，为 A、B，并且一次更新的
20 操作只是对于一个文件 A 的更新，因此数据更新操作还包括如下步骤：将被更
新数据的存储块 M 内本次不更新的其他文件 B 的内容进行复制，再将该文件 B
内容搬移到当前指针更新写入的存储块 N 相应位置上。

综上所述，本发明通过控制区的合理划分和标识记录，有效地实现了对闪
存片数据块读写信息的管理，并将数据块写入和擦除动作分别完成，大大提高
25 了数据读写的速度，而且还可均匀地使用闪存片个数据块，延长其使用寿命。

说明书附图

存储块编号	数据区	状态标识位组
#1	空	坏的
#2	空	空
	空	空
	空	空
	空	空
#N	空	空

图 1

存储块编号	数据区	索引位组信息	逻辑地址指针组	状态标识位组
#1	空	1	空	坏的
#2	空	1	空	空
	空	1	空	空
	空	1	空	空
	空	1	空	空
#N	空	1	空	空

图 2

5

10

存储块编号	数据区	索引位组信息	逻辑地址指针组	状态标识位组
#1	空	1	空	坏的
#2	空	1	空	空
	Data 2	0	2	忙
	空	1	空	空
#N	空	1	空	空
	空	1	空	空

15

图 3

20

存储块编号	数据区	索引位组信息	逻辑地址指针组	状态标识位组
#1	空	1	空	坏的
#2	空	1	空	空
	1st Data 2	0	2	释放
		0	3	
#3	2nd New Data 2	1	空	忙
#N	空	1	空	空
	空	1	空	空

25

图 4

5

10

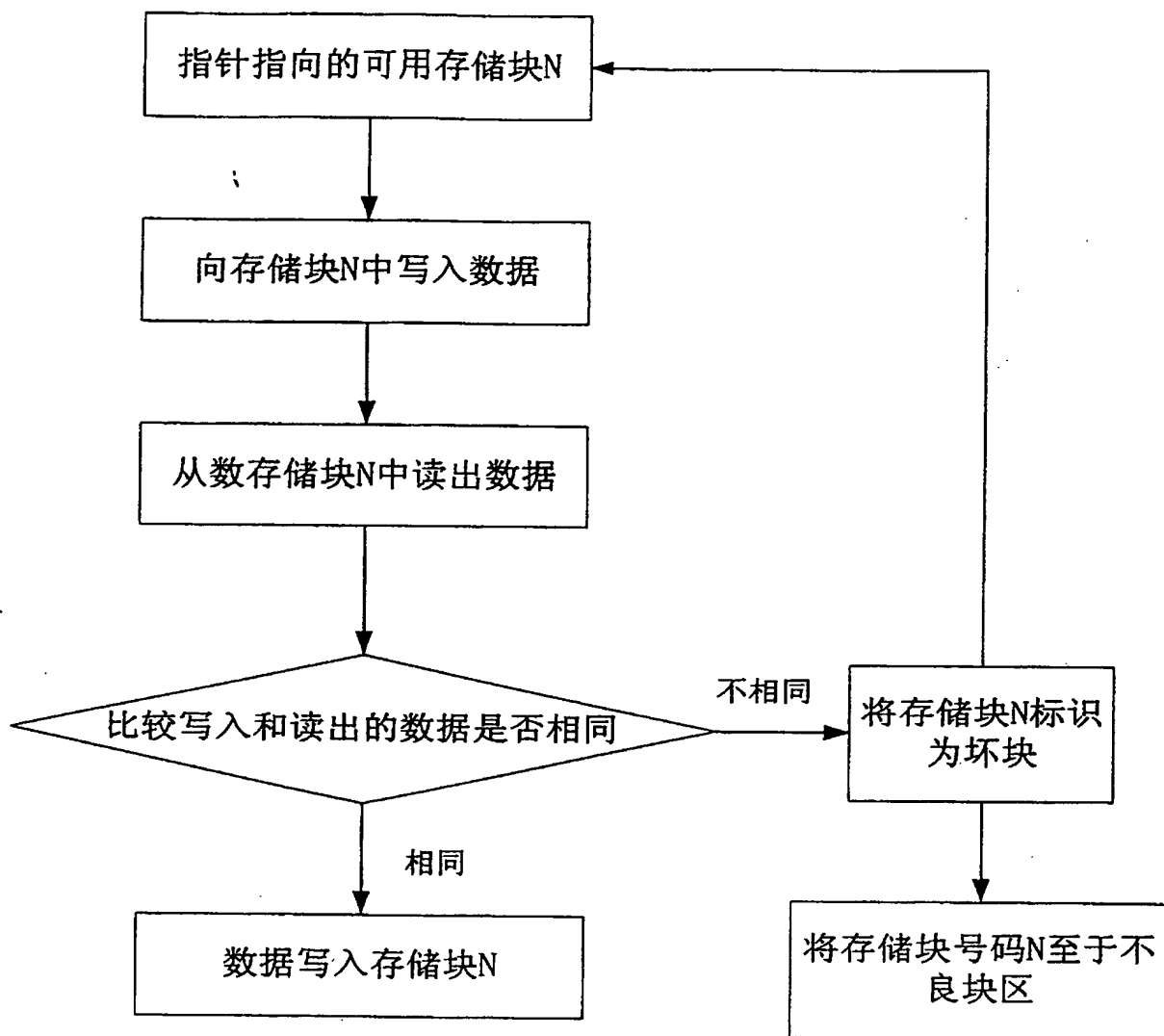


图 5

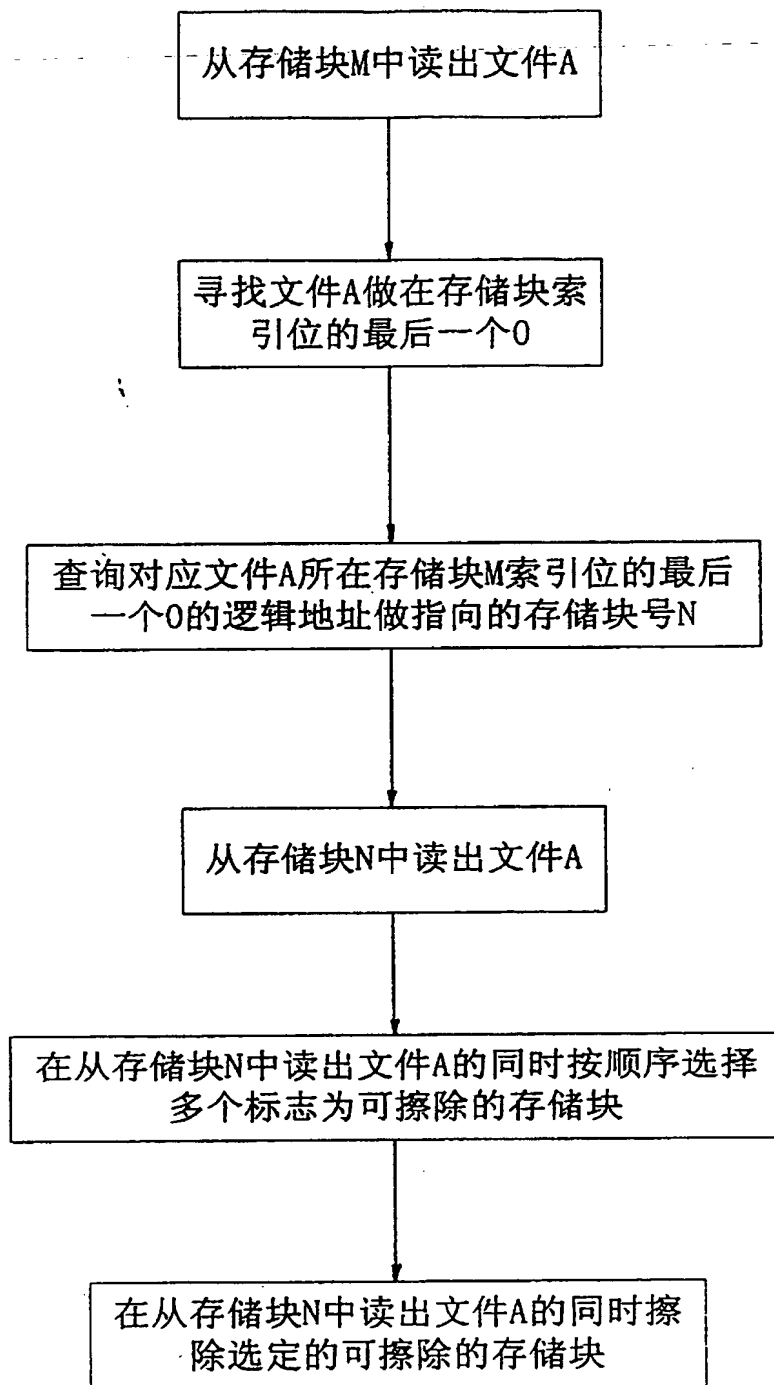


图 6

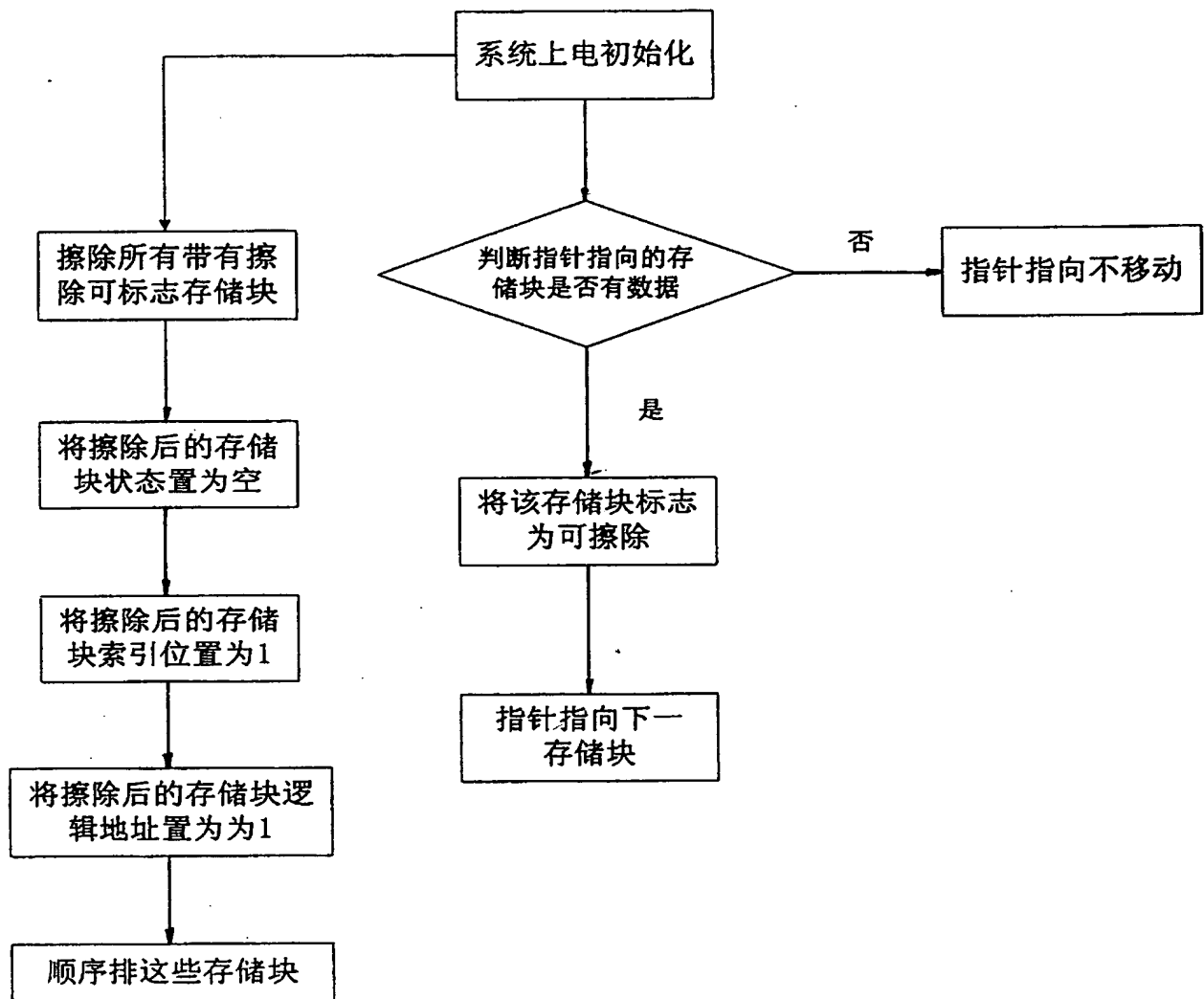


图 7

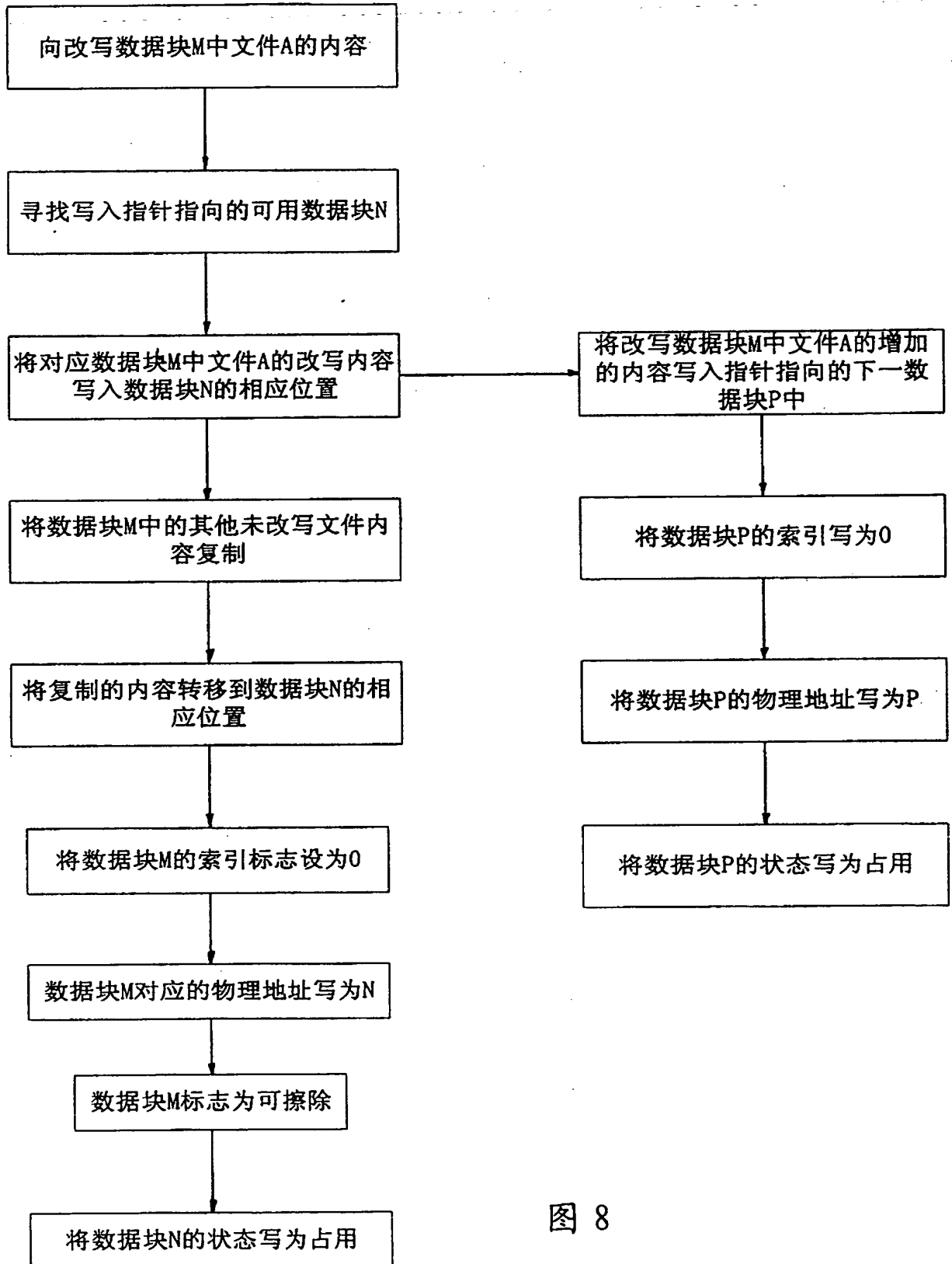


图 8